Simulation Visualization and Data Management

A key component of our effort is support for postprocessing and visualization tools, which include the Griz finite-element postprocessor, the Mili Data Input/Output Library, and an associated data tool XmiliCS. These tools are heavily used by LLNL analysts to interpret data from a variety of simulation codes such as DYNA3D, ParaDyn and NIKE3D.

Griz is our primary tool for visualizing finite-element analysis results computed on 3-D unstructured meshes. Griz also calculates and displays auxiliary result quantities for a variety of codes. Griz provides modern 3-D visualization techniques such as isocontours and isosurfaces, cutting planes, vector field display, and particle traces. Griz also incorporates the ability to animate all representations over time. Mili provides the primary data path between analysis codes and Griz. Xmilics is a utility that consolidates results from large parallel simulations into a single data stream suitable for Griz.

Project Goals

The primary goal of this project is to provide ongoing support for postprocessing tools and add new capabilities to support unique programmatic requirements. The "free-node" capability that was implemented this year is an example of a capability that was added to support one project team, yet is subsequently available to support all programs.

Relevance to LLNL Mission

Postprocessing tools such as Griz and Mili provide important user interfaces for our simulation capabilities and are critical elements in

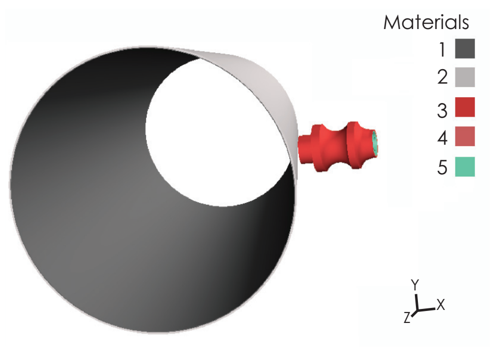


Figure 1. Projectile penetrating a tube (initial state).

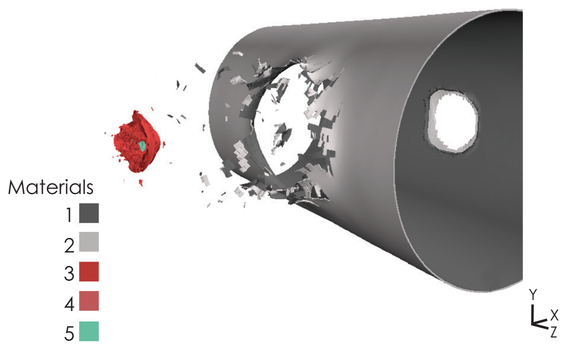


Figure 2. Tube penetration and deformed penetrator.

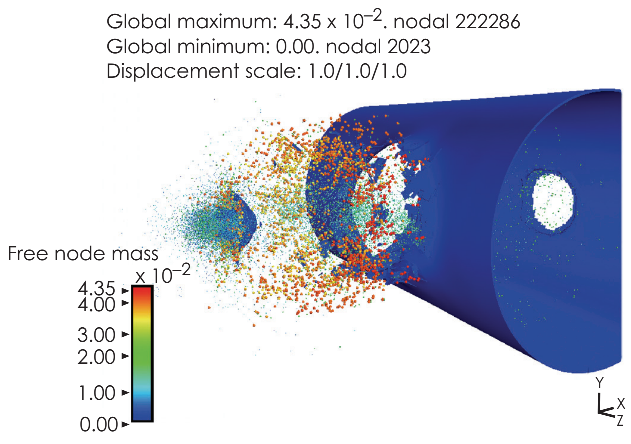

Global maximum: 4.35 x 10^-2. nodal 222286
Global minimum: 0.00. nodal 2023
Displacement scale: 1.0/1.0/1.0
Free node mass
x 10^-2
4.35
4.00
3.00
2.00
1.00
0.00
Y
X
Z

Figure 3. Projection simulation with free node visualization.

For more information contact
Ivan R. Corey
(925) 423-3271
corey3@llnl.gov

our simulation tool suite. Ultimately, the analyst synthesizes analytical conclusions through visual interrogation of the simulation output.

FY2005 Accomplishments and Results

We made significant progress in adding capabilities to Griz and Mili to manage and display surface objects. This is an essential step for facilitating the writing of boundary conditions and data onto surfaces from the analysis programs. The entire GUI implementation and database work has been completed. We are in the process of performing integration tests.

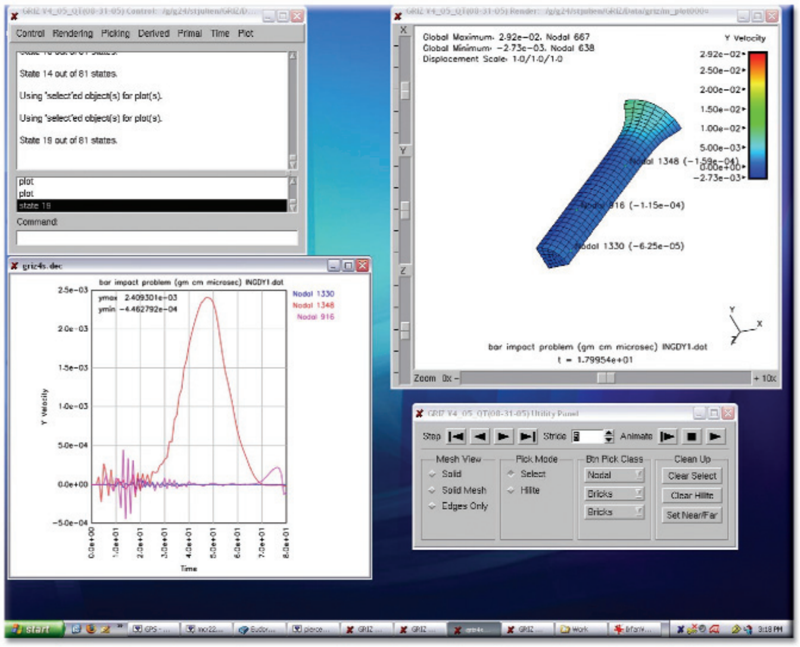
We made a variety of enhancements to support visualization for a project team modeling fragmentation and debris from hypervelocity impacts. Griz can display the nodes of elements having material that failed, and thus constitute a debris field. These nodes are called “free nodes,” and are rendered as spheres. The size of the sphere is based on the mass of the node and a user-defined scaling factor. We can also plot by element volume.

Figures 1 to 3 are snapshots taken from the initial and end states of a simulation, showing the fragmentation and the added representational power of the free node visualization.

We implemented a new capability in Griz to display the extreme minimum or maximum values per element for a selected result quantity. When activated, the values for all states are evaluated and the requested extreme displayed on the deformed mesh at a user-selected time state. We also completed a project to support validation efforts at LANL by providing a plotting capability in Griz for strain data from time-history databases. We were able to speed the calculations by reducing the required amount of data-file reading by a factor of eight.

We continued to make improvements in the software engineering practices of the Griz tool suite. We added several enhancements to the configuration scripts. We also added a feature to allow users to select a specific version of the software at runtime. The code user can query the code for configuration parameters such as build time, data, platform, configuration options, and run-time options.

We made significant progress in converting the Griz window management software from the old Motif library to the modern Qt toolkit. Qt is a comprehensive open source C++ framework for building GUIs. It includes a class library and tools for cross-platform deployment. Figure 4 is a sample screen shot of Griz/Qt.



```
State 14 out of 81 states.
Using 'select'ed object(s) for plot(s).
Using 'select'ed object(s) for plot(s).
State 19 out of 81 states.
```

```
plot
plot
state 19
```

```
Global Maximum: 2.92e-02, Nodal 667
Global Minimum: -2.73e-03, Nodal 638
Displacement Scale: 1.0/1.0/1.0
```

```
ymax   2.409301e-03
ymin  -4.462792e-04
```

FY2006 Proposed Work

We will continue to provide support and enhancements for our user-base at LLNL, LANL, and at several other sites. A major revision of the Mili I/O library is being evaluated for its potential to facilitate more flexible output from our simulation codes and broader data interrogation and fusion by our analysts. These revisions will also be weighed with an eye toward leveraging the state of the art, parallel visualization capabilities created by LLNL’s VisIt tool, to enable our analysts to productively use larger models for our most demanding simulation tasks.

Figure 4. Griz GUI implemented with the Qt toolkit.